Modelling and diagnosis of dynamic systems from timed observations

I. Fakhfakh, M. Le Goc, L. Torres, C. Curt

in : ECAI Workshop - Diagnostic REAsoning: Model Analysis and Performance (DREAMAP), Montpellier, France, 27/08/2012

Modeling and diagnosis of dynamic systems from timed observations

Ismail Fakhfakh [1] and **Marc Le Goc** [2] and **Lucile Torres** [2] and **Corinne Curt** [1]

Abstract. This paper proposes the use of the Timed Observation theory as a powerful framework for model-based diagnosis. In fact, this theory provides a global formalism for modeling a dynamic system (TOM4D), for characterizing and computing diagnoses of the system under investigation

1 INTRODUCTION

In the last two decades model-based diagnosis has been an important research area where numerous new methodologies and formalisms have been proposed, studied and experimented ([3] and [9]). This is motivated by the practical need for ensuring the correct and safe functioning of large complex systems. These frameworks have been created (i) to provide semantics for the diagnosis problem solving, (ii) to analyze the properties and to characterize the diagnosis reasoning and (iii) to give modeling principles.

In dynamic systems, the observation is timed unlike in static systems where the observations are given at only one point of time. This is restrictive in several fields. The extension of the problem poses many problems with the existing approaches. Since (Reiter, 1987), most of the frameworks are based on the logic formalism. Despite of the important contributions in the domain of temporal logics, there is still a difficulty to take into account the time of the observations in the diagnosis reasoning. Later, the Discret Event System formalism has been used to diagnose dynamic systems [1]. One basic difficulty that arises is then the definition of the observations. Cordier [4] proposes to slice off the flow of the measurements into temporal windows to define the observations within these slices and to compute the diagnosis incrementally using the observations of the successive slices. This approach is applied only to D.E.S and is seldom used in real cases. One of the problems with this kind of approach is to define the size of the slices so that the relevant observations can be perceived: there is no a priori reason for the observations to be synchronized with the slicing algorithm. In other words, the slicing algorithm can mask pertinent observations and, within a slice, the observations must be ordered to be taken into account in a model. These difficulties are classical with discrete time systems. To avoid these problems, Le Goc [8] proposes to define observations time-stamped with clocks in time continuous. The Timed Observation Theory of Le Goc [8] provides a general mathematical framework for modeling dynamic processes from timed data. The application of this framework to diagnosis has given birth to a modeling methodology for diagnosis TOM4D (Timed Observation Theory for Diagnosis). The aim of the modeling methodology is to provide an efficient diagnosis based on models built at the same abstraction level as these of the experts.

In this paper, after a brief presentation of the Timed Observation Theory and the TOM4D method (Sections 2 and 3), we show how TOM4D supports the modeling of complex physical systems. In sections 4 and 5, we show how the models can be used to characterize the diagnosis and we demonstrate that the diagnosis can be computed easier using the TOM4D models (section 6). We apply the modeling approach and the diagnosis algorithm to an hydraulic system. Finally, Section 7 provides conclusions and proposes some perspectives to this work.

2 THEORY OF TIMED OBSERVATIONS

Le Goc's Timed Observation Theory extends Shannon's Theory of Communication to timed data and offers a unique frame for Markov Chains and Poisson Theories. It also extends the Logical Theory of Diagnosis to timed observations. This theory considers that the timed messages of a serie are written in a database by a program called a Monitoring Cognitive agent (MCA), which monitors a dynamic sytem. A dynamic system is a process $Pr(t)$=$\{x_1(t)$, $x_2(t)$, ..., $x_n(t)\}$ defined as an arbitrary set made of time functions $x_i(t)$ defined on the real set denoted $\Re$ (i.e. $\forall$ t $\in \Re$, $x_i(t) \in \Re$).

This theory defines a timed observation in the following way [8]. Given a set $Pr(t)$=$\{x_1(t)$, $x_2(t)$, ..., $x_n(t)\}$ of time functions the evolution of which are observed by a program Θ; let $X = \{x_1$, x_2, ..., $x_n\}$ be the corresponding set of variable names; let $\Delta = \bigcup_{\forall x_i \in X} \Delta_{x_i}$ each $\Delta_{x_i} =\{\delta_1^i$, δ_2^i, ..., $\delta_m^i\}$ being a set of constants denoting the possible values for x_i; let Γ=$\{t_k\}_{t_k \in \Re}$ be a set of arbitrary time instants.

Definition 1 (A Timed Observation). *A timed observation $o(t_k) \equiv (\delta_j^i, t_k)$, made by a program θ when observing a time function $x_i(t)$ at time $t_k \in \Gamma$, is the assignation of the values v=x_i, δv=δ_j^i and t=t_k to a predicate $\Theta(v, \delta v, t)$ so that: $\Theta(x_i, \delta_j^i, t_k)$.*

Conceptually, the θ program applies the spatial segmentation principle: a value δ_j^i is assigned to a variable x_i whenever the value of its corresponding time function $x_i(t)$ enters in a range $[\psi^i, \psi^{i+1}[$, where ψ^i is a threshold for $x_i(t)$ (i.e. $\psi^i \in \Re$). This means that the values are assigned to the variables with a program (or a human) the basic specification of which is the following (cf. [7] for examples of more complex spatial segmentation algorithms):

$\forall$ k $\in \mathbb{N}$, $x_i(t_k) \geq \psi^i \wedge x_i(t_{k-1}) < \psi^i \Rightarrow o(t_k) \equiv (\delta_j^i, t_k) \wedge t_k \in \Gamma$

In practice, each time t_k the predicate $\Theta(x_i, \delta_j^i, t_k)$ is assigned, the program θ (or a human) writes a couple (δ_j^i, t_k) in a database, a datalog or a simple document. As a consequence, to any timed observation o(t_k)$\equiv$ (δ_j^i, t_k) corresponds an assigned predicate $\Theta(x_i, \delta_j^i,$

[1] IRSTEA, 3275 route de Cézanne - CS 40061, Aix-en-Provence, France
[2] LSIS, Aix-Marseille Univ, 13397 Marseille, France

t_k). [8] shows that this predicate can always be interpreted as the "Equal" predicate so that: $\Theta(x_i, \delta_j^i, t_k) \equiv \text{Equal}(x_i, \delta_j^i, t_k) \Leftrightarrow x_i(t_k) \in [\psi^i, \psi^{i+1}[$. Such an assigned predicate is often represented in the expert's language under the form of the assignation of the value δ_j^i to the variable x_i at t_k: $x_i(t_k)=\delta_j^i$. The value δ_j^i can therefore then be considered as a symbol denoting the range $[\psi^i, \psi^{i+1}[$. This leads to define the notion of class of observations.

Definition 2. *An observation class* $C^i=\{(x_i, \delta_j^i), (x_{i+1}, \delta_{j+1}^{i+1}), ..., (x_{i+n}, \delta_{j+n}^{i+n}\}$ *is a set of couples* (x_i, δ_j^i) *associating a variable* x_i*, eventually unknown, with a constant* δ_{j+k}^{i+k}*.*

In other words, an observation class C_i associates variables $x_i \in X$ with constants $\delta_j^i \in \Delta_{x_i}$. This leads to the following property:

Proposition 2.1. *Each timed observation* $o(t_k) \equiv (\delta_j^i, t_k)$ *corresponds to an occurrence of an observation class* $C_i = \{(x_i, \delta_j^i)\}$*.*

In practical applications, the observation classes are usually defined as a singleton of the form $C_i = \{(x_i, \delta_j^i)\}$. These definition allow defining a modeling methodology for diagnosis.

3 MODELING APPROACH FOR DIAGNOSIS : TOM4D

TOM4D is a modeling methodology for dynamic systems focused on timed observations. The objective of this methododology is to produce a suitable model for dynamic process diagnosis from timed observations and experts' a priori knowledge. TOM4D relies on the idea that experts use an implicit model to both formulate the knowledge about the process and diagnose it. It is a multi-model approach that combines CommonKads templates [11] with the conceptual framework proposed in [12] and the tetrahedron of states (T.o.S), [10], [2]. These elements are merged according to the Timed Observations Theory [8].

The TOM4D methodology is based on the notion of observation class $C_i = \{(x_i, \delta_j^i)\}$ and associates the variable x_i of each observation class C_i with one and only one component c_i. This means that the values δ_j^i a variable x_i can take over time is the result of a couple $(\theta(\Delta_{x_i}), x_i(t))$ made with a program $\theta(\Delta_{x_i})$ that observes the evolutions of a time function $x_i(t)$ and write a timed observation $o(t_k) \equiv (\delta_j^i, t_k)$ whenever a predicate $\Theta(x_i, \delta_j^i, t_k)$ is assigned. In other words, $x_i(t)$ is the signal provided by some sensors associated with a component c_i. This allows to organize the available knowledge about a process $Pr(t)$ according to (i) a Perception Model $PM(Pr(t))$ defining the process as an arbitrary set made of time functions $x{-}i(t)$ and its operating goals and its normal and abnormal behaviors, (ii) a Structural Model $SM(Pr(t))$ defining the components of the process and their relations, (iii) a Functional Model $FM(Pr(t))$ defining the relations between the values of the process variables (i.e. their definition domain) with a set of mathematical functions, and (iv) a Behavior Model $BM(Pr(t))$ defining the timed observation classes firing the evolutions of the time functions of $Pr(t)$.

Figure 1 describes the three main steps of the TOM4D modeling process: Knowledge Interpretation, Process Definition and Generic Modeling. The aim of this process is to produce a coherent generic model $M(Pr(t)) = < PM(Pr(t)), SM(Pr(t)), FM(Pr(t)), BM(Pr(t)) >$ from the available knowledge and data.

The Knowledge Interpretation step uses a CommonKADS template to interpret and to organize the available knowledge about a dynamic system. This knowledge is provided by a knowledge source (an expert, a set of documents, etc) and when possible,

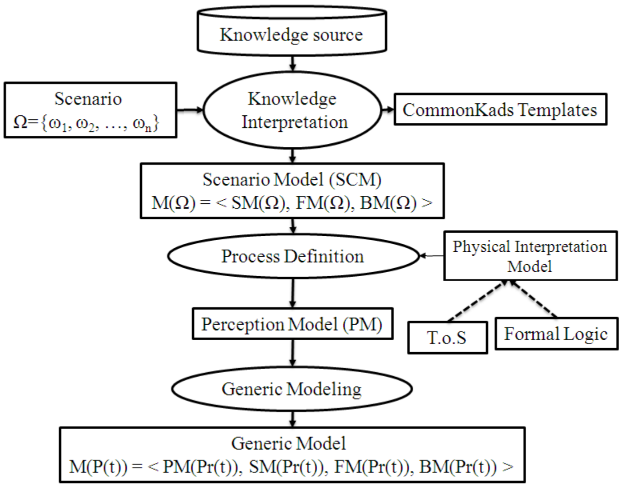

Figure 1. TOM4D Modeling Process

at least one scenario. This first step aims at producing a scenario model $M(\Omega) =< SM(\Omega), FM(\Omega), BM(\Omega) >$ of the system that is coherent with the available knowledge about its evolution over time. This model is used in the Process Definition step to provide a definition of the process under the form of a perception model $PM(Pr(t))$. This is made with the use of the tetrahedron of states to provide a physical dimension to each variable of the process and with the use of formal logic to define its operating goals and its normal and abnormal behaviors. The aim of this step is to control the way the semantics of the available knowledge is introduced in the model to avoid the potential representation errors. The Perception Model $PM(t)$ defined, the Generic Modeling step aims at defining an abstract representation of the dynamic system where the different terms of the available knowledge are reified through a set of relations. This paper being focused on the use of the resulting model $BM(Pr(t))$, the interested reader is invited to see [9], [5] or [6] for further details about TOM4D.

A TOM4D behavior model $BM(Pr(t))$ describes the possible sequences of observation classes that can occur and therefore the discernible states between them.

Definition 3. *A behavior model* $BM(Pr(t))$ *of a dynamic process* $Pr(t)$ *is a 3-tuple* $< S, C, \gamma >$ *where:*

- $S = \{s : X \to \Delta | s(x_i) = \delta, x_i \in X, \delta \in \Delta\}$ *is a set of functions which characterize the discernible states of the process* $Pr(t)$*,*
- C *is a set of observation classes, where an observation class associated with a variable* $x_i \in X$ *is a set* $C^i = \{(x_i, \delta) | \delta \in \Delta^{x_i}\}$ *containing only one element (i.e. a singleton),*
- $\gamma : S \times C \to S$ *is a function of discernible state transition.*

Given a sequence $\omega = \{o(k)\}$ of observation class occurrence $o(k) \equiv (\delta^i, t_k)$, a transition from a discernible state s_i to the discernible state s_j is triggered when:

- there is an occurrence $o(k) \equiv (\delta^i, t_k)$ of class C^i in ω;
- the current state $s(t)$ of the finite state machine implementing $BM(Pr(t))$ is the discernible state s_i (i.e. $s(t) = s_i$);
- there exists an assignment $\gamma(C^k, s_i) = s_j$.

The observation classes being singletons, an occurrence of an observation class (i.e. a timed observation $o(k) \equiv (\delta^y, t_k)$) corresponds to the assignation of a particular value δ^y to a variable x_i of $Pr(t)$.

4 SEQUENTIAL BINARY RELATIONS

The important point is that a state transition in a finite state machine implementing a TOM4D behavior model $BM(Pr(t))$ can occur if and only if there exist two assignations $s_i = \gamma(C^x, s_{k-1})$ and $s_k = \gamma(C^y, s_i)$ in $BM(Pr(t))$.

Definition 4. *Given a TOM4D behavior model $BM(Pr(t)) =< S, C, \gamma >$, a sequential binary relation $r(C^x, C^y, s_i)$ between two observations classes C^x and C^y, labelled with a discernible state s_i, exists iff: $\exists s_{k-1}, s_i, s_k \in S, s_i = \gamma(C^x, s_{k-1}) \wedge s_k = \gamma(C^y, s_i)$.*

A sequential binary relation between two observation classes $r(C^x, C^y, s_i)$ is an oriented (sequential) relation between two observation classes $C^x = \{(x, \delta^x)\}$ and $C^y = \{(y, \delta^y)\}$ that is linked with a discernible state s_i. This latter can correspond to the current state of a finite state machine implementing a TOM4D behavior model $BM(Pr(t))$ after observing an occurrence $C^x(t_k) = (\delta^x, t_k)$ of the "input" observation class C^x and before observing the occurrence $C^y(t_{k+1}) = (\delta^y, t_{k+1})$ of the "output" observation class C^y.

The γ function defines then the possible sequential relations between two observation classes:

Proposition 4.1. *Two assignments $s_i = \gamma(C^x, s_{k-1})$ and $s_k = \gamma(C^y, s_i)$ define a sequential binary relation $r(C^x, C^y, s_j)$ between two classes C^x and C^y labelled with a discernible state s_i.*

In other words, a TOM4D behavior model $BM(Pr(t)) =< S, C, \gamma >$ specifies a graph between the set C of observation classes. This graph is used to control the diagnosis reasoning.

A class graph C-Graph is a set $G_C = \{..., r_i(C^x, C^y, s_{x,y}), ...\}$, $i = 1...n$, of sequential binary relations of the form $r(C^i, C^o, s_{io})$ between an input observation class C^i and an output observation class C^o labelled with a discernible state s_{i_o}. The C-Graph is built from a TOM4D generic behavior model generated with the following algorithm.

Algorithm: Generate-C-Graph $G_C = \{r_i\}$
input: a behavior Model $BM(Pr(t)) =< S, C, \gamma >$
output: a C-Graph $G_C = \{r_i\}, r_i \equiv r(C^x, C^y, s_i)$

1.$G_C = \Phi$
2.$\forall s_i \in S$
2.1.$\exists s_n, s_m \in S$,
$s_n = \gamma(s_i, C_x) = s_n \wedge s_m = \gamma(s_n, C^y)$
$\Rightarrow G_C = G_C \cup r(C^x, C^y, s_i)$;
3.$Return G_C$

The C-Graph G_C describes the complete process behavior in terms of observation class. This means that a path in this graph describes a particular behavior of the process. Such a path correspond to a suite of discernable states in the behavior model $BM(Pr(t))$. So looking for a particular suite of discernable states in $BM(Pr(t))$ corresponds to look for a particular path in the associated C-Graph G_C:

Definition 5. *A class path P^C is a sub-graph of a C-Graph G_C made with a suite $P^C = (r_{i,i+1})$, $i = 1...n$ of n sequential binarys relation $r_{i,i+1}$ of the form $r(C^i, C^{i+1}, s_{i,i+1})$.*

In other words, the general form of a class path P^C is the following: ($r_1(C^{i_1}, C^{i_2}, s_{i_1,i_2})$, $r_2(C^{i_2}, C^{i_3}, s_{i_2,i_3})$, ..., $r_n(C^{i_n}, C^{i_{n+1}}, s_{i_n,i_{n+1}})$).

Because the timed observations provided by a MCA $\Theta(X, \Delta)$ are the occurrences of the observation classes of the set C of a TOM4D behavior model $BM(Pr(t))$, it is simpler to look for a class path in the C-Graph and then to look for the corresponding state path, rather that trying to directly build the suite of states from the suite of observations. This idea is the basis of the proposed diagnosis algorithm.

5 DIAGNOSING WITH C-GRAPHS

According to The timed Observation Theory [8], the timed observations are provided by a MCA $\theta(X, \Delta)$ that assumes the online supervision of a dynamic process $Pr(t)$. Diagnosis is performed starting from a sequence $\omega = \{o(t_k)\}$ of timed observations and a TOM4D process model $M(Pr(t))$. It consists in explaining the timed observations of ω written by MCA $\Theta(X, \Delta)$ during a period $[t_0, t_n]$.

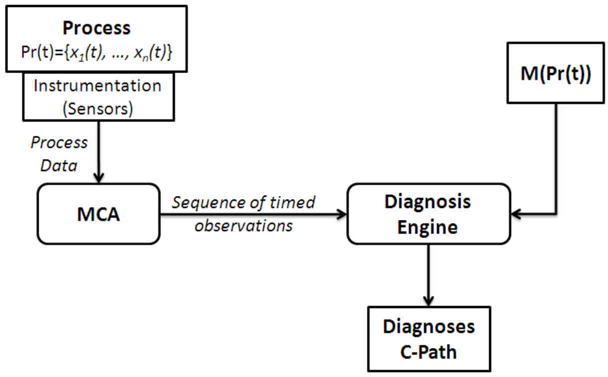

Figure 2. Diagnosis Engine

Consequently, the diagnosis aims at generating the minimal set D of class paths P^C that are compatible with the timed observations of ω (cf. Figure 2) and the C-Graph derived from the behavior model $BM(Pr(t))$ of the TOM4D process model $M(Pr(t))$.

Definition 6 (Diagnosis Definition)**.** *Given a C-Graph $G_C = \{..., r(C^x, C^y, s_i), ...\}$ and a suite $\omega = \{o(t_0), ..., o(t_n)\}$ of $n+1$ timed observations recorded during the period $[t_0, t_n]$, a diagnosis at time $t \in [t_0, t_n]$ is the minimal set $D(t) = \{P^C\}$ of class paths P^C that are consistent with G_C and ω.*

$$(\omega, G_C) \rightarrow D(t_n) \tag{1}$$

The algorithm of computing the minimal set D of class path P^C from a C-graph G_C and a sequence ω of timed observations is made with a loop on each timed observation $o(k) \in \omega$ and acts with three main steps: (i) remove the paths of D that are no more coherent with $o(k)$, (ii) extend each path in the resulting set D with the right sequential relations from G_C and (iii) initialize the set D when it is empty (at the first loop or if there are no more paths that are coherent with ω. The algorithm also uses three functions: "$obsClassOf(o)$" to get the class of a timed observation, "$rightestRelationOf(P)$" to get the right most sequential binary relations of a class path and "$rightRelations(r(C^i, C^o, s_{io}), G_C)$" to get the set of sequential binary relation corresponding to the successor of a particular sequential binary relation $r(C^i, C^o, s_{io})$ in a C-Graph G_C.

Algorithm: Generate-Class-Path
input : a C-Graph G_C and a sequence $\omega = \{o(t_k)\}$ of n timed observations
Output : a set D of class paths consistent G_C and ω

1.$D \leftarrow \{\phi\}$
//Loop on the timed observations of ω
//$\Gamma(\omega)$ is the set of time-stamp of the timed observations of ω
2.$\forall t_k \in \Gamma(\omega)$
//Compute the set C of the classes occured at t_k
2.1.$\forall o(t_k) \equiv (\delta^i, t_k) \in \omega, ObsClassOf(o(t_k)) \in C$
//Compute D for the set C at time t_k
2.2.$D = computeD(D, G_C, C)$
3.$ReturnD$

Algorithm: computeD
input : a set D of C-Path, a C-Graph G_C and a set C of observation classes
Output : the upated set D

1.$d \leftarrow \{\phi\}$
//Loop on the observation classes of C
2.$\forall c \in C$
2.1.$d \leftarrow d \cup computeCPaths(D, G_C, c)$
3.$D \leftarrow d$
//If D is empty, initialise D with C and G_C
4.$D = \{\phi\} \Rightarrow D \leftarrow initCPath(G_C, C)$
5.$ReturnD$

Algorithm: computeCPaths
input : a set D of C-Paths, a C-Graph G_C and an observation class c
Output : the updated set D

1.$D_1 = \{\phi\}$ //Working set of C-Path
//Remove from D the paths that are not compatible with c
2.$\forall P \in D$
2.2.$r(C^i, C^o, s_{io}) \leftarrow rightestRelationOf(P)$
2.3.$C^o = c \Rightarrow D_1 = D_1 + P$
3.$D \leftarrow D_1$ //D contains the C-Paths compatible with c
//Extends each path of D with the right sequential relations
4.$D_1 = \{\phi\}$ //Reset the working set D_1
4.$\forall P \in D$
4.1.$r(C^i, C^o, s_{io}) \leftarrow rightestRelationOf(P)$
//Get the relations from G_C
4.2.$R = rightRelations(r(C^i, C^o, s_{io}), G_C)$
4.3.$\forall r \in R$
4.3.1.$P_1 = P + r$ //Create a new extended path for P
4.3.2.$D_1 = D_1 + P_1$ //Add the new path in D_1
5.$ReturnD_1$

The next section illustrates this algorithm on the (simple) device of Figure 3 studied in [3]. It is to note that this algorithm can easily be extended to simultaneous timed observations that can occur in large and complex systems. In other hand, the lack of timed observations leads the algorithm to remove the C-Paths that are no more consistent with the suite of timed observations. It can also be extended to use the functional model $FM(Pr(t))$ to distinguish between a true lack of timed observation and an inconsistency between the sequence of timed observations and the behavior model $BM(Pr(t))$.

6 APPLICATION

[3] describes the example with the following terms: the system is formed by a pump P which delivers water to a tank TA via a pipe PI; another tank CO is used as a collector for water that may leak from

Algorithm: initCPath
input : a C-Graph G_C and a set C of observation classes
Output : a set D of sequential binary relations consistent with C

1.$\forall c \in C$
1.1$\forall r(C^i, C^o, s_{io}) \in G_C$,
1.2$C^o = c \Rightarrow \{r(C^i, C^o, s_{io})\} \in D$
2.$ReturnD$

the pipe. The pump is always on and supplied of water. The pipe PI can be ok (delivering to the tank the water it receives from the pump) or leaking (in this case we assume that it delivers to the tank a low output when receiving a normal or low input, and no output when receiving no input). The tanks TA and CO are simply receive water. We assume that three sensors are available (see the eyes in Figure 3): $flow_p$ measures the flow from the pump, which can be normal (nrm_p), low (low_p), or zero (zro_p); $level_{TA}$ measures the level of the water in TA, which can be normal (nrm_{ta}), low (low_{ta}), or zero (zro_{ta}); $level_{CO}$ records the presence of water in CO, either present (pre_{co}) or absent (abs_{co}).

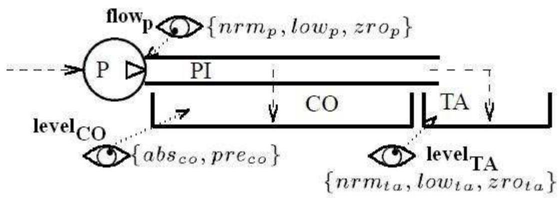

Figure 3. Hydraulic system

According to the TOM4D methodology, the system is a hydraulic process $Pr(t) = \{x_1(t), x_2(t), x_3(t)\}$ made with three variables (cf. the hydraulic T.o.S): $x_1(t)$ is a volume variable, $x_2(t)$ and $x_3(t)$ are two outflow variables. The analysis of the system description shows that $x_2(t)$ represents a normal outflow and $x_3(t)$ represents an abnormal outflow corresponding to water leakage. Table 1 shows the variable-value association and the physical interpretation of the variables. The corresponding set of observation classes is given in Table 2 and the discernible states are provided in Table 3. The reader interested with the application of the TOM4D methodology on this example is invited to refer to [5].

Variables x	Physical interpretation	value interpretation	Abstract value δ_i
x_1	$Volume$	$normal,$	2,
		$low,$	1,
		$zero$	0
x_2	$normal$ $outflow$	$normal,$	2,
		$low,$	1,
		$zero$	0
x_3	$abnormal$ $outflow$	$presence,$	2,
		$absence$	1

Table 1. Variable-Value Association for the Hydraulic System

in : ECAI Workshop - Diagnostic REAsoning: Model Analysis and Performance (DREAMAP), Montpellier, France, 27/08/2012

$C_1^1 = \{(x_1,0)\}$ $\quad C_2^1 = \{(x_1,1)\}$ $\quad C_3^1 = \{(x_1, 2)\}$

$C_1^2 = \{(x_2,0)\}$ $\quad C_2^2 = \{(x_2,1)\}$ $\quad C_3^2 = \{(x_2,2)\}$

$C_1^3 = \{(x_3,1)\}$ $\quad C_2^3 = \{(x_3,2)\}$

Table 2. Timed Observation Classes

$States$	x_1	x_2	x_3	$States$	x_1	x_2	x_3
s_0	0	0	1	s_1	1	0	1
s_2	2	0	1	s_4	1	1	1
s_5	2	1	1	s_8	2	2	1
s_9	0	0	2	s_{10}	1	0	2
s_{11}	2	0	2	s_{13}	1	1	2
s_{14}	2	1	2	s_{17}	2	2	2

Table 3. The set of discernible states for the Hydraulic System

Figure 4 shows a graphical representation of the behavior model $BM(Pr(t))$ of the hydraulic system. The "$Generate{-}C{-}Graph$" algorithm of section 4 produce the C-Graph G_C of Figure 5.

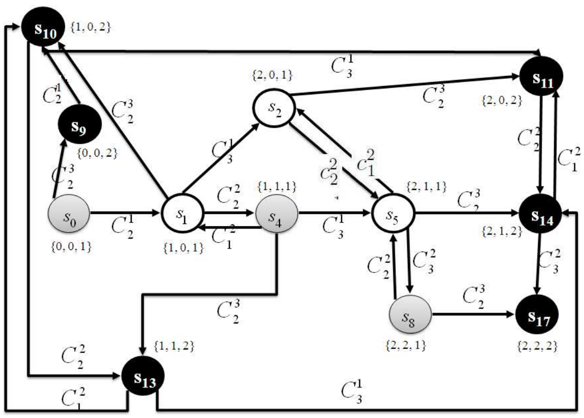

Figure 4. behavior Model of the hydraulic system

To illustrate the "$Generate - Class - Path$" algorithm of the previous section, let us consider the following sequence of timed observations: $\omega = \{\ o_{x_2}(t_0) \equiv (1, t_0), o_{x_3}(t_1) \equiv (2, t_1), o_{x_2}(t_2) \equiv (0, t_2)\ , o_{x_1}(t_3) \equiv (2, t_3)\}$ (We consider that $t_i \leq t_{i+1}$). According to the table 2, the observation class associated to the first timed observation $o_{x_2}(t_0)$ of ω is C_2^2. The set D being empty, the two first steps of the algorithm do nothing but the third step initializes D with the Algorithm *"initCPath"* that is to say finds the set of binary relation that are of the form r(C^0, C_2^2, s_{i_0}) so that D = { {r(C_2^1 ,C_2^2, s_1)}, {r(C_1^2 ,C_2^2, s_1)}, {r(C_1^2 ,C_2^2, s_2)}, {r(C_3^1 ,C_2^2, s_2)}, {r(C_3^2 ,C_2^2, s_8)}, {r(C_1^2 ,C_2^2, s_{11})}, {r(C_2^3 ,C_2^2, s_{11})}, {r(C_3^1 ,C_2^2, s_{11})}, {r(C_1^2 ,C_2^2, s_{10})}, {r(C_2^3 ,C_2^2, s_{10})}, {r(C_2^1 ,C_2^2, s_{10})}}.

The observation class of the second timed observation $o_{x_3}(t_1) \equiv (2, t_1)$, $o_{x_2}(t_2)$ being C_2^3, the next step of the algorithm removes the paths of D that are no more coherent with $o_{x_3}(t_1)$ and extends the rest of paths with the right sequential relations from G_C (cf. Figure

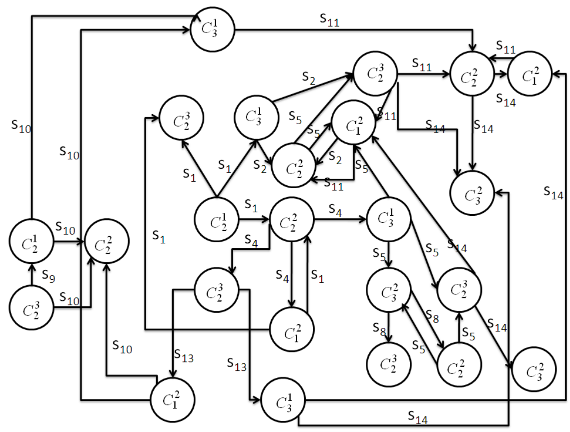

Figure 5. C-Graph of the hydraulic system

5) so that $D = \{\ \{r(C_2^1, C_2^2, s_1), r(C_2^2, C_2^3, s_4)\}, \{r(C_1^2, C_2^2, s_1), r(C_2^2, C_2^3, s_4)\}, \{r(C_1^2, C_2^2, s_{11}), r(C_2^2, C_2^3, s_{14})\}, \{r(C_1^2, C_2^2, s_2), r(C_2^2, C_2^3, s_5)\}, \{r(C_3^1, C_2^2, s_2), r(C_2^2, C_2^3, s_8)\}, \{r(C_3^2, C_2^2, s_5), r(C_2^2, C_2^3, s_5)\}\ \}$. Doing so, the algorithm finds only two C-Paths that are consistent with all the timed observations of ω (cf. Fig 6). The dark circle means that the new observation class is inconsistent with the defined C-Path (there is no relation between the last observation class and the new observation class).

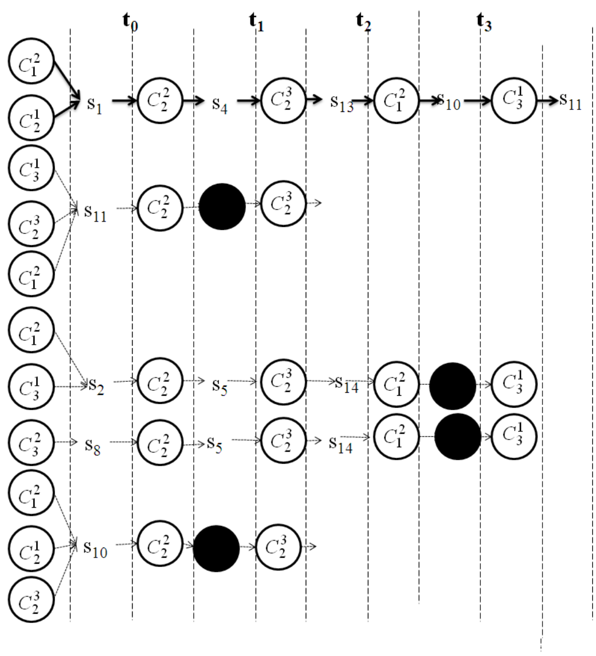

Figure 6. P^C consistent with the ω and BM

$D(t) = \{P^{C_1}, P^{C_2}\} = \{\ \{r_0(C_2^1, C_2^2, s_1), r_1(C_2^2, C_2^3, s_4), r_2(C_2^3, C_1^2, s_{13}), r_3(C_1^2, C_3^1, s_{10})\}, \{r_0(C_1^2, C_2^2, s_1), r_1(C_2^2, C_2^3, s_4), r_2(C_2^3,$

C_1^2, s_{13}), $r_3(C_1^2$, C_3^1, $s_{10})\}\}$ and the state Path corresponding is S-Path = $\{s_1$, s_4, s_{13}, s_{10}, $s_{11}\}$. The interpretation of the results with the behavior model shows that the system passed from the ok mode (the grey states in Figure 4) : states (s_1, s_4) to leaking mode (the dark states in Figure 4) : states (s_{13}, s_{10}, s_{11}).

7 CONCLUSION

This paper proposes an algorithm to diagnose dynamic systems modeled with the TOM4D methodology according to the Theory of Timed observations of [8]. This alogorithm is a preliminary work since we have not exploited all the potentialities of the theory. In particular, this algorithm does not consider the lack of timed observations that can occur in large and complex systems. An extension is under consideration with the idea to use the function model $FM(Pr(t))$ to discriminate between a true lack and an inconsistency.

On other hand, with large and complex systems, the impossibility to define a global behavior model obliges to model the behavior in a decompositional way with the description of the behaviors of each component of the system. Another extension to the proposed algorithm aims at computing the diagnosis locally for each component before merging the local diagnosis to get a global diagnosis. In the D.E.S. approaches, the diagnoses are merged using the events which are common with the local diagnosis. According to the TOM4D methodology, the observations classes are not common between two components because, by construction, each variable x_i is associated with one and only one component c_i. Consequently, the idea is to use the functional model $FM(Pr(t))$ to define the relation between the observation classes and to merge the local diagnosis.

ACKNOWLEDGEMENTS

The authors would like to thank the Provence - Alpes - Côte d'Azur Région and FEDER (Fonds Européen de DEveloppement Régional) for their funding.

REFERENCES

[1] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, 1999.

[2] L. Chittaro, G. Guida, C. Tasso, and E Toppano, 'Functional and teological knowledge in the multi-modeling approach for reasoning about physical systems: A case study in diagnosis', in *IEEE Transactions on Systems, Man, and Cybernetics, 23(6)*, pp. 1718–1751, (1993).

[3] L. Console, C. Picardi, and M. Ribaudo, 'Diagnosing and diagnosability analysis using pepa', in *14th European Conference on Artificial Intelligence*, (2000).

[4] M.-O. Cordier and A. Grastien, 'Exploiting independence in a decentralised and incremental approach of diagnosis', in *20th International Joint Conference on Artificial Intelligence*, (2007).

[5] I. Fakhfakh, M. Le Goc, L. Torres, and C. Curt, 'Modeling and diagnosis characterization using timed observation theory', in *7th International Conference on Software and Data Technologies, ICSoft*, (2012).

[6] I. Fakhfakh, M. Le Goc, L. Torres, and C. Curt, 'Modeling dynamic systems for diagnosos: Pepa/tom4d comparison', in *14th International Conference on Enterprise Information Systems, ICEIS*, (2012).

[7] M. Le Goc, 'SACHEM, a real-time intelligent diagnosis system based on the discrete event paradigm', *Simulation*, **80**(11), 591–617, (2004).

[8] M. Le Goc, 'Notion d'observation pour le diagnostic des processus dynamiques : application sachem et à la découverte de connaissances temporelles', in *Université Aix-Marseille III - Faculté des Sciences et Techniques de Saint Jérôme*, (2006).

[9] M. Le Goc, E. Masse, and C. Curt, 'Modeling processes from timed observations', in *3rd International Conference on Software and Data Technologies, ICSoft*, (2008).

[10] R. Rosenberg and D. Karnopp, 'Introduction to physical system dynamics', in *McGraw-Hill, Inc. New York, NY, USA.*, (1983).

[11] Th. Schreiber, J. M. Akkermans, A. A. Anjewierden, R. de Hoog, N. R. Shadbolt, W. Van de Velde, and B. J. Wielinga, 'Publication, knowledge engineering and management the commonkads methodology', *MIT Press*, (2000).

[12] C. Zanni, M. Le Goc, and C. Frydmann, 'A conceptual framework for the analysis, classification and choice of knowledge-based system', in *International Journal of Knowledge-based and Intelligent Engineering Systems, 10*, pp. 113–138. Kluwer Academic Publishers, (2006).